

4. Noise Covariance Matrices

Noise Covariance Matrices

Claim Term	CMU's Construction	Marvell's Construction
noise covariance matrices	noise statistics used to calculate the ‘correlation-sensitive branch metrics.’	covariance matrices of signal samples (where the signal samples include noise).

'839 Patent Claims 11, 16, 19, 23
'180 Patent Claim 6

CMU Brf. at 27

Marvell Brf. at 27-32

- The Dispute
 - ▶ Does “noise covariance matrices” have its ordinary meaning (Marvell) or has it been re-defined in the patent (CMU)?

Claim Language

- “noise covariance matrices” used to calculate “correlation sensitive branch metrics”

6. A method of detecting a sequence that exploits a correlation between adjacent signal samples for adaptively detecting a sequence of symbols through a communications channel having intersymbol interference, comprising:

- performing sequence detection on a plurality of signal samples using a plurality of correlation sensitive branch metrics;
- outputting a delayed decision on a transmitted symbol;
- outputting a delayed signal sample;
- adaptively updating a plurality of **noise covariance matrices** in response to the delayed signal samples and the delayed decisions;
- recalculating the plurality of correlation sensitive branch metrics from the **noise covariance matrices** using subsequent signal samples; and
- repeating steps (a)–(e) for every new signal sample.

15
-continued
 $y_k(m', m) = P(x_k = m | x_{k-1} = m')$
 $f(z_k | x_{k-1} = m', x_k = m, z_{k-1}, z_{k-2}, \dots, z_{k-L})$
 $= P(x_k = m | x_{k-1} = m') \cdot (2\pi)^{-0.5} e^{-0.5 M_M}$
 $\alpha_k(m) = \sum_{m''=0, M-M-1} [\gamma_k(m', m) \alpha_{k-1}(m'')]$
 For $k = K-1, K-2, \dots, 0$
 $\beta_k(m) = \sum_{m'', i=0, M-1} [\gamma_{k+1}(m, m'') \beta_{k+1}(m'')]$
 For $k = 0, 1, \dots, K$
 $\lambda_k(m) = \alpha_k(m) \beta_k(m)$
 $\delta_k(m', m) = \alpha_{k-1}(m') \gamma_k(m', m) \beta_k(m)$

Thus, the branch metric, as denoted by the second Equation x, is computed exactly the same way as the process of Equations 6, 9, 10, 11, and 13. When the process is Gaussian, the branch metric can be computed using Equation 13 and the arrangements described in 3A and 3B.

The generalization of the case described above BCJR algorithm can be made for any other soft or hard output algorithm defined on a trellis or a graph convolutional (or other dynamic) system. The placement and computation of the metric function can be substituted by the metric computation as described in Equations 6, 9, 10, 11, 13 and FIGS. 3A and 3B.

While the present invention has been described in conjunction with preferred embodiments thereof, many modifications and variations will be apparent to those of skill in the art. For example, the present invention can be used to detect a sequence that exploits the correlation between adjacent signal samples for adaptively detecting a sequence of symbols through a communications channel having intersymbol interference. The foregoing description and the following claims intended to cover all such modifications and variations.

What is claimed is:

- A method of determining branch metric values for a detector, comprising:
 receiving a plurality of time variant signal samples having one of signal-dependent correlated noise, and both signal dependent noise associated therewith;
 selecting a branch metric function at a certain point;
 applying the selected function to the signal samples to determine the metric values.

- The method of claim 1, wherein the function is selected from a set of signal metric functions.

- The method of claim 1, wherein it is selected from a group consisting of a Viterbi Viterbi detector, a Generalized Viterbi detector, and a decision feedback detector.

- A method of detecting a sequence that exploits a correlation between adjacent signal samples for adaptively detecting a sequence of symbols through a communications channel having intersymbol interference, comprising:

- performing sequence detection on a plurality of signal samples using a plurality of correlation sensitive metrics;

- outputting a delayed decision on a transmitted symbol.

Specification

- Covariance Matrix C_i used in Correlation-Sensitive Branch Metric Calculation:

With this notation, the general correlation-sensitive metric is:

$$M_i = \log \det \frac{C_i}{\det c_i} + N_i^T C_i^{-1} N_i - \mu_i^T c_i^{-1} \mu_i \quad (13)$$

US 6,201,839 B1

In the derivation of the branch metrics (8), (13) no assumptions were made on the exact Viterbi-type architecture, that is, the metrics can be applied to any Viterbi-type algorithm such as PRML, FETS/DE, RAM-BER, etc., etc.

FIG. 3A illustrates a block diagram of a branch computation circuit 48 that computes the metric M branch of a trellis, as in Equation (13). Each branch trellis requires a circuit 48 to compute the metric M . A logarithmic circuit 50 computes the first term right hand side of (13):

$$\left(\log \det \frac{C_i}{\det c_i} \right)$$

and a quadratic circuit 52 computes the second term right hand side of (13) (i.e. $N_i^T C_i^{-1} N_i - \mu_i^T c_i^{-1} \mu_i$). The through the circuits 50 and 52 represent the outputs of the Viterbi-like detector 50. A sum circuit 53 of FIG. 4B sums the outputs of the circuits 50 and 52.

As stated above, the covariance matrix is given

$$C_i = \begin{bmatrix} \alpha_i & \beta_i \\ \beta_i & \gamma_i \end{bmatrix}$$

Using standard techniques of signal processing, shown that:

$$\det \frac{C_i}{c_i} = \alpha_i + \beta_i^T c_i^{-1} \beta_i$$

This ratio of determinants is referred to as η_i^2 , i.e.

$$\eta_i^2 = \frac{\det C_i}{\det c_i} = \alpha_i + \beta_i^T c_i^{-1} \beta_i$$

It can be shown by using standard techniques of processing that the sum of the last two terms of (13) output of the circuit 53, is

$$\beta_i = N_i^T C_i^{-1} N_i - \mu_i^T c_i^{-1} \mu_i$$

$$= \frac{(N_i^T N_i)^2}{\det c_i}$$

Where the vector N_i is ($L+1$)-dimensional and is given by

$$N_i^T = [1 \quad w_2 \quad w_3 \quad \dots \quad (w_L+1)]^T$$

$$= \begin{bmatrix} 1 \\ -w_1^T c_i \end{bmatrix}$$

Equations (17), (18) and (16) (the circuit 52) can be merged as a tapped-delay line as illustrated in FIG. 3B, circuit 52 has 1 delay circuits 54. The tapped-delay implementation shown in FIGS. 3A and 3B is also good as a moving-average, feed-forward, or finite-response filter. The circuit 48 can be implemented by type of filter as appropriate.

The covariance matrix C_i corresponds to the sequence (trellis path) and is used to compute the metrics (13) in the subsequent steps of the Viterbi-like algorithm.

US 6,201,839 B1

11 **12**

$\hat{C}'(\Theta, +, -) = \beta \hat{C}(\Theta, +, -) + (1 - \beta) N_i N_i^T$ **(26)**

$= \begin{bmatrix} 0.4755 & -0.189 \\ -0.189 & 0.7620 \end{bmatrix}$

'839 Patent 6:66-7:3

'839 Patent 7:25-29

'839 Patent 11:1-10

As stated above, the covariance matrix is given as:

$$C_i = \begin{bmatrix} \alpha_i & \beta_i \\ \beta_i & \gamma_i \end{bmatrix} \quad (14)$$

TABLE I

Receiving Parameters

Symbol rate	1000 symbols/sec
Symbol duration	0.001 sec
Symbol thickness	0.0001 sec
Symbol cross-track correlation width	0.0001 sec
Symbol length	0.0001 sec
Symbol gradient filter	0.0001 sec
Symbol per symbol	0.0001 sec
Symbol width (symbol period)	0.0001 sec
Symbol length (symbol period)	0.0001 sec
Symbol width (symbol period)	0.0001 sec

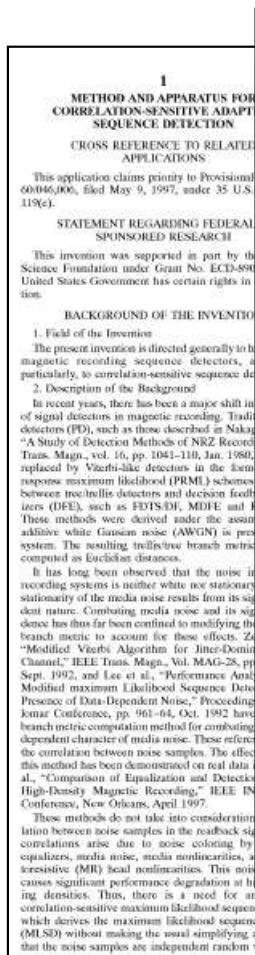
Table I: Recording Parameters

The table provides parameters for a recording system. The symbol rate is 1000 symbols/sec, symbol duration is 0.001 sec, symbol thickness is 0.0001 sec, symbol cross-track correlation width is 0.0001 sec, symbol length is 0.0001 sec, symbol gradient filter is 0.0001 sec, symbol per symbol is 0.0001 sec, symbol width (symbol period) is 0.0001 sec, symbol length (symbol period) is 0.0001 sec, and symbol width (symbol period) is 0.0001 sec.

Symbol separation of 3.5s. This recording density corresponds to a symbol density of 2.5 symbols/W30. FIG. 1

Specification

- First reference to “the noise covariance matrices”
- Implies the ordinary definition



SUMMARY OF THE INVENTION

•

Because the noise statistics are non-stationary, the noise sensitive branch metrics are adaptively computed by estimating **the noise covariance matrices** from the read-back data. These covariance matrices are different for each branch of the tree/trellis due to the signal dependent structure of the media noise. Because the channel characteristics in magnetic recording vary from track to track, these matrices are tracked on-the-fly, recursively using past samples and previously made detector decisions.

'839 patent 2:15-23

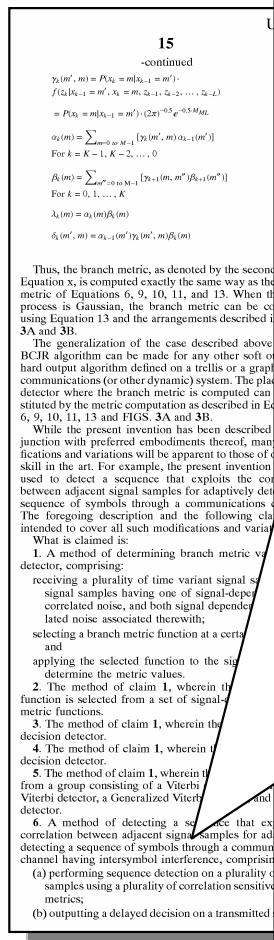
CMU's "i.e." Argument Fails

- CMU argues: “By using the abbreviation ‘*i.e.*,’ Latin for ‘that is,’ the CMU patents equated the term ‘noise covariance matrices’ with ‘noise statistics.’”
CMU Brf. at 28
- Fails for Two Reasons:
 1. English Usage: “*i.e.*, like *that is*, typically introduces a rewording or clarification of a statement that has just been made or of a word that has just been used”
Webster’s Dictionary of English Usage (1989) (Surreply Exh. 1)
 - example: “to update my contact information, *i.e.* to update my address”
 - “the adaptive nature, *i.e.*, the data dependent nature, of the circuit 52”
‘839 Patent 8:9-10
 2. Federal Circuit: “when read in the context of the entire [] patent, the reference to ‘saccharides (*i.e.*, sugars)’ does not constitute a definition of ‘saccharides.’”

Pfizer, Inc. v. Teva Pharmaceuticals, USA, Inc., 429 F.3d 1364 (Fed. Cir. 2005).

CMU's "i.e." Argument Fails

- CMU reads out “covariance matrices” by reading in redundant language (“correlation-sensitive branch metrics”)



6. A method of detecting a sequence that exploits a correlation between adjacent signal samples for adaptively detecting a sequence of symbols through a communications channel having intersymbol interference, comprising:

- performing sequence detection on a plurality of signal samples using a plurality of correlation sensitive branch metrics;
- outputting a delayed decision on a transmitted symbol;
- outputting a delayed signal sample;
- recalculating the plurality of correlation sensitive branch metrics from the noise covariance matrices using subsequent signal samples; and
- repeating steps (a)-(e) for every new signal sample.

See '839 Patent Claims 11, 16, 19, 23; '180 Patent Claim 6

CMU's "i.e." Argument Fails

- Re-defining “noise covariance matrices” as “noise statistics” removes “covariance” and “covariance matrices” from the claims
 - ▶ “[a] claim construction that gives meaning to all the terms of the claim is preferred over one that does not do so.”
Merck & Co., Inc. v. Teva Pharm. USA, Inc., 395 F.3d 1364, 1372 (Fed. Cir. 2005).
 - ▶ “covariance” and “covariance matrices” have well-known meanings in engineering and statistics
 - ▶ “covariance” and “covariance matrices” are used independently in other claims

See '839 Patent Claim 10.

CMU's "i.e." Argument Fails

- Patents use “noise covariance matrices” separately from “noise statistics”

Because the noise statistics are non-stationary, the noise sensitive branch metrics are adaptively computed by estimating the noise covariance matrices from the read-back data. These covariance matrices are different for each branch of the tree/trellis due to the signal dependent structure of the media noise. Because the channel characteristics in mag-

‘839 Patent 2:15-23

Computing the branch metrics in (10) or (13) requires knowledge of the signal statistics. These statistics are the mean signal values m_i in (12) as well as the covariance matrices C_i in (13). In magnetic recording systems, these

'839 Patent 8:24-27

CMU's "i.e." Argument Fails

- “noise statistics”: used by all branch metrics
- “covariance matrices”: used by only “correlation-sensitive branch metric”

Specific expressions for the branch metrics that result under different assumptions on the noise statistics are next considered.

$$\prod_{i=1}^L \frac{f(x_{i+1}, \dots, x_{i+K})}{f(x_i, \dots, x_{i+K-1})}$$

The factorial form of equation (5) is suitable for Viterbi-like dynamic programming decoders.

‘839 Patent 5:56-58

Variance dependent branch metric.

$$M_i = \log \sigma_i^2 + \frac{N_i^2}{\sigma_i^2} = \log \sigma_i^2 + \frac{(r_i - m_i)^2}{\sigma_i^2} \quad (10)$$

$$d_1, \dots, d_L \text{ are independent}$$

Correlation-sensitive branch metric. In the most general case, the correlation length is L>0. The leading and trailing ISI lengths are K and K' respectively. The noise is now

‘839 Patent 6:15-34

With this notation, the general correlation-sensitive metric is:

$$M_i = \log \det \frac{C_i}{\det c_i} + N_i^T C_i^{-1} N_i - n_i^T c_i^{-1} n_i \quad (13)$$

tional signal paths are factored as

is

‘839 Patent 6:66-7:4

CMU's "All Embodiments" Argument Fails

- CMU: "Marvell's proposed construction improperly narrows the claims so that they do not encompass all embodiments disclosed in the patents that use 'noise covariance matrices' to compute the correlation-sensitive branch metric."

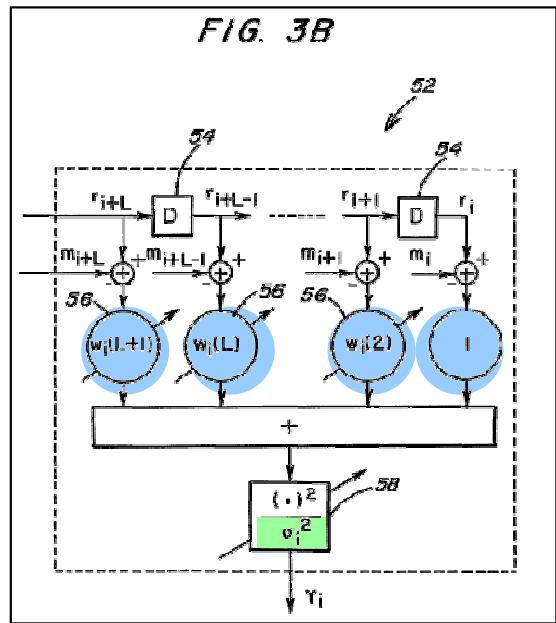
CMU Brf. at 31

- Fails for two reasons:
 1. "[A] claim need not cover all embodiments."
Intamin v. Magnetar, 483 F.3d at 1337 (Fed. Cir. 2007)
 2. Marvell's construction covers all of the embodiments that use covariance matrices

CMU's "All Embodiments" Argument Fails

- CMU: "Specifically, Marvell's construction excludes the tapped-delay line FIR filter embodiment of Figure 3B"
 - "In this embodiment, the noise statistics used to calculate the branch metrics are the variance σ^2 and the vector of weights w_i "

CMU Brf. at 31
McLaughlin Decl. (CMU) at ¶ 18



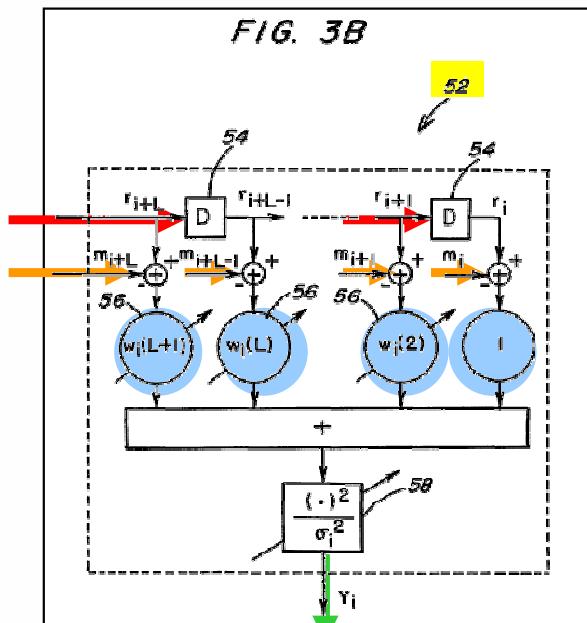
the data dependent nature, of the circuit 52. The weights w_i and the value σ_i^2 can be adapted using three methods. First, w_i and σ_i^2 can be obtained directly from Equations (20) and (16), respectively, once an estimate of the signal-dependent covariance matrix C_i is available. Second, w_i and σ_i^2 can be calculated by performing a Cholesky factorization on the inverse of the covariance matrix C_i . For example, in the $L_i D_i^{-1} L_i^T$ Cholesky factorization, w_i is the first column of the Cholesky factor L_i and σ_i^2 is the first element of the diagonal matrix D_i . Third, w_i and σ_i^2 can be computed directly from the data using a recursive least squares-type algorithm. In the first two methods, an estimate of the covariance matrix is obtained by a recursive least squares algorithm.

839 Patent
8:10-23

- The patent describes three methods for calculating σ^2 and w_i .
- Marvell's construction covers "the first two methods" (which are the only embodiments described using covariance matrices)

CMU's “All Embodiments” Argument Fails

- The third method (no covariance matrix) corresponds to claims that calculate a “weight w_i ” (Group III Claims)
 - These claims do not use “noise covariance matrices”



Equations (17), (18) and (16) (the circuit 52) can be implemented as a tapped-delay line as illustrated in FIG. 3B. The

20. A branch metric computation circuit for generating a branch weight for branches of a trellis for a Viterbi-like detector, wherein the detector is used in a system having Gaussian noise, comprising:

a logarithmic circuit having for each branch an input responsive to a branch address and an output;
 a plurality of arithmetic circuits each having a first input responsive to a plurality of signal samples, a second input responsive to a plurality of target response values, and an output, wherein each of the arithmetic circuits corresponds to each of the branches;

a sum circuit having for each branch a first input responsive to said output of said logarithmic circuit, a second input responsive to said output of said arithmetic circuit, and an output.

21. The circuit of claim 20 wherein said branch metric computation circuit is a tapped-delay line circuit with adaptive weight.